Vol 8. No. 2 June, 2015
African Journal of Computing & ICT

www.ajocict.net

Towards the Development of a Time-Out Multiple C-R CAPTCHA Framework Using Integrated Mathematical Modeling

J.O. Okesola
School of Computing
University of South Africa
South Africa
tunji_okesola@yahoo.co.uk

Longe O. B.
Department of Computer Science & Mathematics
Adeleke University, Ede, Osun State, Nigeria.
longeolumide@fulbrightmail.org

A.P. Obi
42d Earlham grove
London
E7 9AW.
prcs_obi@yahoo.com

ABSTRACT

The internet has suffered from large forms of insecurity ranging from scamming, hacking and theft of information. Lately the use of CAPTCHAs has become a common security tool for authentication and authorization. However CAPTCHAS has suffered from certain vulnerabilities in the context of the simplicity offered by the challenge-response scenario and its timing which leaves room for improvement. This paper proposes a Time-Out Multiple Challenge-Response (C-R) CAPTCHA Framework that Utilizes Mathematical Modelling as a basis for overcoming some of the challenges faced by current CAPTCHA Systems. Our approach ensures security during the authorization and authentication process.

Keywords: Time-Out System, Challenge-Response, Security, Authorization, Authentication and CAPTCHA

African Journal of Computing & ICT Reference Format:
J.O. Okesola, O.B. Longe & A.P. Obi (2015). Towards the Development of a Time-Out Multiple C-R CAPTCHA Framework Using Integrated Mathematical Modeling. Afr J. of Comp & ICTs. Vol 8, No. 2. Pp 145-154.

I. INTRODUCTION

CAPTCHA stands for "Completely Automated Public Turing Test to Tell Computers and Human Apart". They are slightly distorted images, challenges or tests administered automatically over networks that can distinguish between people and machines (automated script) and thus protect web services from abuse by programs disguising as human users. [2][4][5]. Captcha is a security poser administered over the internet to monitor the authenticity of each usage of resources on the internet. They are challenges or tests meant to be easily solved by humans while remaining too hard to be economically solved by computers. For example, humans can read distorted text as displayed in figure 1 but current computer programs cannot. Captcha are usually a single challenge response system but there are studies that present the double challenge response system called 2RC –(dualistic captcha 2010).

CAPTCHAs are used by Yahoo, Hotmail, PayPal, Microsoft, TicketMaster, Register.com, Google and many other popular sites to prevent and protect services against automatic script attacks and automated registrations.

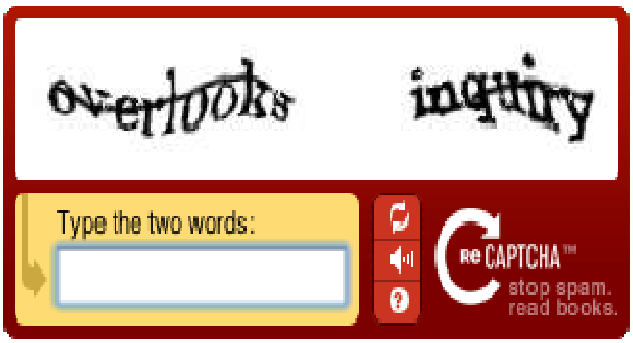

Fig 1: Distorted Image Example

CAPTCHAs work because no computer program can currently read distorted text as well as a human being. Cyber security involves protecting information by preventing, detecting and responding to attacks through various mechanisms. Measures currently in use include Intrusion Detection Systems (IDSs) and Intrusion Prevention Systems (IPSs). These protective measures are to combat the risks involved in the application of computers and internet technology infrastructures. Other measures taken against threats to resources on the internet include the use of cryptography (decryption and encryption), password, signatures, iris recognition, Captcha, etc. [1].

2. CAPTCHAs IN PERSPECTIVE

The term CAPTCHA was coined in 2000 by Luis von Ahn, Manuel Blum, Nicholas Hopper and John Langford of Carnegie Mellon University [4][7]. Previous studies showed CAPTCHA system as an alphanumeric single challenge response system. However, malicious attackers have been able to use automated script to answer the challenge with the aid of OCR (Optical Character Recognition) while sophisticated attackers employ social engineering (relay attacks) to address the problem. To be effective, a CAPTCHA must be difficult enough to discourage script attacks by raising the computation and/or development cost of been broken to an unprofitable level. At the same time, it must be easy enough to solve so as not discourage humans from using the service but at the same time maintain some good levels of security. (recaptcha.com). The single challenge system is such that the text pops in slightly distorted and the user is expected to retype the popped up text or audio on the space provided, in the case where the user fails to type in the text correctly, it changes the text and represents another text to the user and it goes on for as long as the network can take, the problem with this is that it takes up too much time that could be used for something else and in the case where the user is a non-human, it keeps allowing access to the site or information. Over the years there has not been a complete work on the remedy to this problem though work is on-going on some other ways to correct this anomaly and making the captcha more secured.[10][9].

2.1 Applications of CAPTCHAS.

Captchas is applicable to several practical security scenarios some of which are discussed in this section

Preventing Comment Spam in Blogs. Most bloggers are familiar with programs that submit bogus comments, usually for the purpose of raising search engine ranks of some website (wikipedia.org). By using a Captcha, one can ensure that only humans can enter comments on a blog. [14].

Protecting Website Registration. Several companies like Yahoo offer free email services. Until few years ago, most of these services suffered from a specific type of bot attack that would sign up for thousands of email accounts every minute [15]. The solution to this problem was to use captchas to ensure that only humans obtain free accounts. In general, free services should be protected with a captcha in order to prevent abuse by automated scripts. [9]

Protecting Email Addresses From Scrapers.

Spammers crawl the Web in search of email addresses posted in clear text. Captchas provide an effective mechanism to hide your email address from Web scrapers. The idea is to require users to solve a captcha before showing your email address.(www.captcha.net)

Online Polls. In November 1999, there was an online poll asking which was the best graduate school in computer science. As in the case of online polls, IP addresses of voters were recorded in order to prevent single users from voting more than once. However, students at Carnegie Mellon University (CMU) found a way to stuff the ballots using programs that voted for the university thousands of times as the university's score started growing rapidly. The next day, students at Massachusetts Institute of Technology (MIT) wrote their own program and the poll became a contest between voting "bots." MIT finished with 21,156 votes, Carnegie Mellon with 21,032 and every other school with less than 1,000 votes. This explains why the result of any online poll cannot be trusted until when the poll can ascertain that only humans can vote [15]

Preventing Dictionary Attacks. Captchas can also be used to prevent dictionary attacks in password systems. The idea is simple: prevent a computer from being able to iterate through the entire space of passwords by requiring it to solve a captcha after a certain number of unsuccessful logins. This is better than the classic approach of locking an account after a sequence of unsuccessful logins, since doing so allows an attacker to lock accounts at will [12]

Search Engine Bots. It is sometimes desirable to keep Web Pages unindexed to avoid them from being easily located. There is an html tag to prevent search engine bots from reading web pages. However, the tag does not guarantee that bots would not read a web page. Search engine bots, since they usually belong to large companies, respect web pages that don't want to allow them in. However, captchas are needed in order to truly guarantee that bots won't enter a web site [15].

Worms and Spam. Captchas also offer a plausible solution against email worms and spam. A few companies are already marketing this idea of ensuring that a user only accepts emails if they are sure there is a human behind the computer.

2.2 Problem Statement

Current CAPTCHA algorithms are plagued with certain inadequacies due to their simplicity and the easiness of the questions offered by the challenge-response system. Their timing is also flawed and this makes them more vulnerable and calls for improvement.

3. THE TM-MCRS FRAMEWORK

The intention of this research is to strengthen the application of CAPTCHA by introducing an Integrated Mathematical Mechanism with the Time Out System to a Multiple Response Challenge Captcha. The Time-Out Multiple Challenge Response System (TM-MCRS) is proposed as a mechanism where a user is expected to answer two CAPTCHA tests by providing two responses before he/she is granted access to the protected resources. The system times out the user where the user fails to get through the second challenge. The first challenge is the retype design and the second is the integrated mathematical challenge. A program will be developed based on the SHAI algorithm to generate the CAPTCHA. This will be implemented using PHP, Netbeans, and Java. The TM-MCRS is a system where the challenges are administered in a multiple of times. The first being the alphanumeric text challenge where the user is expected to retype just exactly the generated text as it appears, followed by the challenge called the integrated mathematical mechanism where the user is expected to give correct answers to a mathematical challenges for him/her to proceed. The timing out is essential in that it will reduce the chances bots have to input answers to the challenge.

The **TM-MCRS** framework emulates and improve the general classical CAPTCHA procedures for a single challenge system as proposed by [7] are as follow

- Computer generates a test instance.
- Test is shown to the human/bot.
- Human/bot attempts to solve the test.
- Human/bot reports suppose solution to the computer.
- Computer evaluation the submitted solution.
- Computer reports the result of evaluation to the human/bot
- Computer allows access if solution submitted is correct

Our implementation adopts the following guidelines:

Accessibility. Captchas must be accessible. It is solely based on reading text or other visual-perception tasks to prevent visually impaired users from accessing the protected resource. Such captchas may make a site incompatible with Section 508 in the United States. Any implementation of a captcha should allow blind users to get around the barrier, for example, by permitting users to opt for an audio captcha.

Image Security. Captcha images of text should be distorted randomly before being presented to the user. Many implementations of captchas use undistorted text, or text with only minor distortions. These implementations are vulnerable to simple automated attacks. (recaptcha.net)

- **Script Security.** Building a secure captcha code is not easy. In addition to making the images unreadable by computers, the system should ensure that there are no easy ways around it at the script level. An example of insecurities in this respect is where systems bypass the answer to the captcha in plain text as part of the web form. Captcha scripts found freely on the Web are mostly vulnerable to these forms of attacks [14]

Security Even After Wide-Spread Adoption. There are various captchas that would be insecure when a significant number of sites start using them. This attribute is inadequate as true captchas should remain secure regardless of the numbers of websites adopting them.

3.1 Design Issues

This section gives us an insight into the design of our proposed TM-MRC, which is a way of strengthening captcha system and combating cyber-crimes of different sorts. A text challenge is administered once followed by a simple mathematical test (which is the second challenge). A time out is then added to the mechanism to reduce both the time wasted on the system for each access and the time bot has to wait to deduce or solve the captcha system. This implies that once the second challenge is administered the time taken to solve the challenge starts counting expecting that the human user will have to solve the challenge at a given time and of which a bot will have same time to attempt the challenge. In a case where the bot throws the captcha words to spammer especially on some pornographic site, the time taken to solve the captcha will not meet the time that has already started counting on the site where captcha is administered. This reduces the time a bot has to solve a captcha text thereby increasing the security of resources on the site administered with captcha. This is used to screen out bots. The concept of this captcha and the previous CAPTCHAs is motivated by real-world problems faced by internet companies such as Yahoo! and AltaVista. These companies offer free email accounts, intended for use by humans. However, they found that many online vendors were using "bots" - the computer programs that would sign up for thousands of email accounts from which they could send out masses of junk emails.

3.2 Typical Captcha Image Generation

Research has shown that spam bots or users using Optical Character Recognition (OCR) software to read and capture existing texts are able to bypass CAPTCHA tests with a good level of success. The need therefore arise to present a second challenge that contains mathematical elements. This challenge will be administered to check the advances made

by bot and then the time out administered to reduce the number of trials a user or spambot requires to detect the administered challenge. The system automatically generates the challenges - a distorted text with busy background to make it unreadable for OCR - and the simple mathematical calculation all stored in the database which displays a "session has expired" message after the second failed attempt thereby reducing the number of trials a user or spam can make attempts. Below is figure 2 illustrating a typical single captcha challenge response architecture. Figure 3 depicts a Typical CAPTCHA Image Generation Implementation Framework

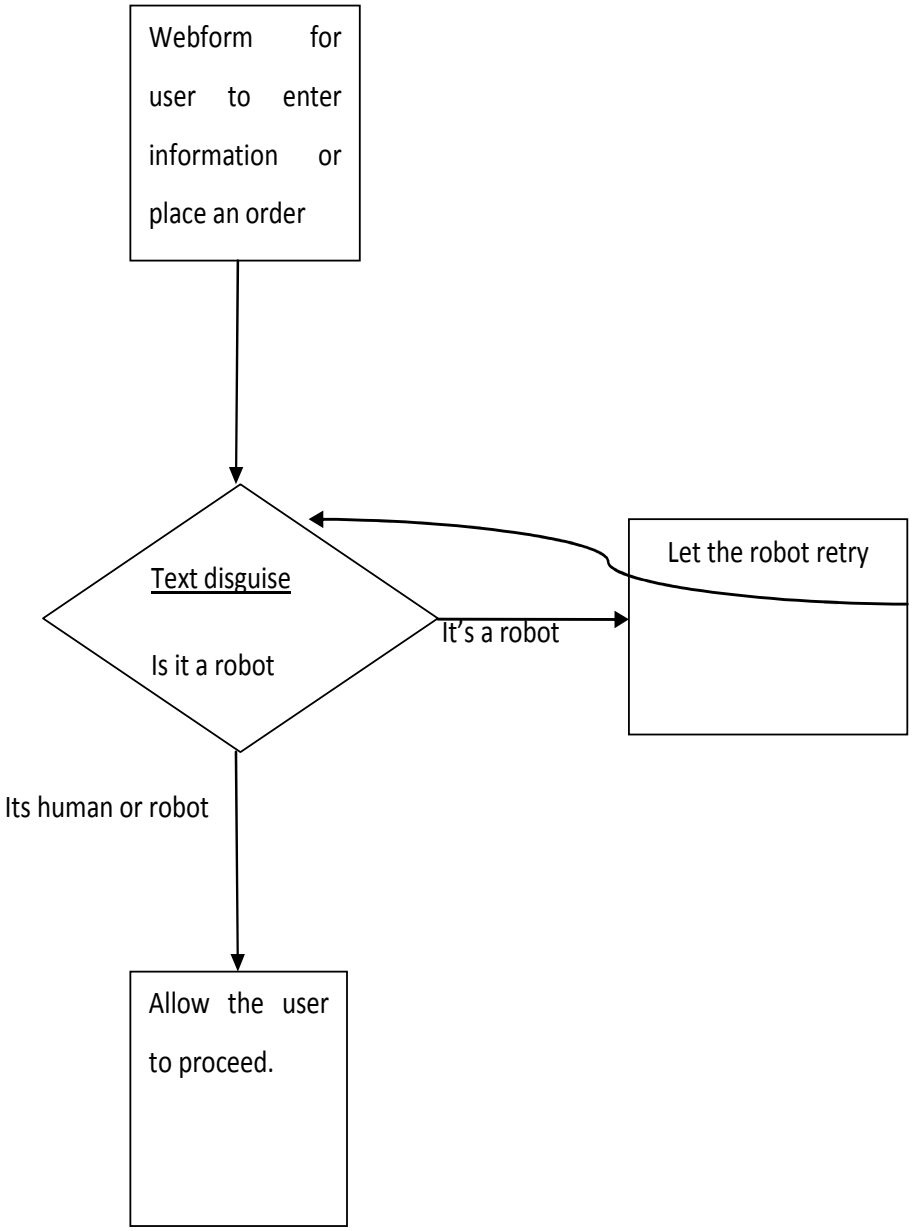

Fig 2. Typical Single Response Captcha Architecture

The general algorithm for a classical CAPTCHA implementation framework described in figure 3 is as follows:

1. Computer generates a test instance;
2. Test is shown to the human/bot;
3. Human/bot attempts to solve the test;
4. Human/bot reports suppose solution to the computer;
5. Computer evaluate the submitted solution; and
6. Computer reports the result of evaluation to the human/bot and allows or blocks access to a resource based on the result.

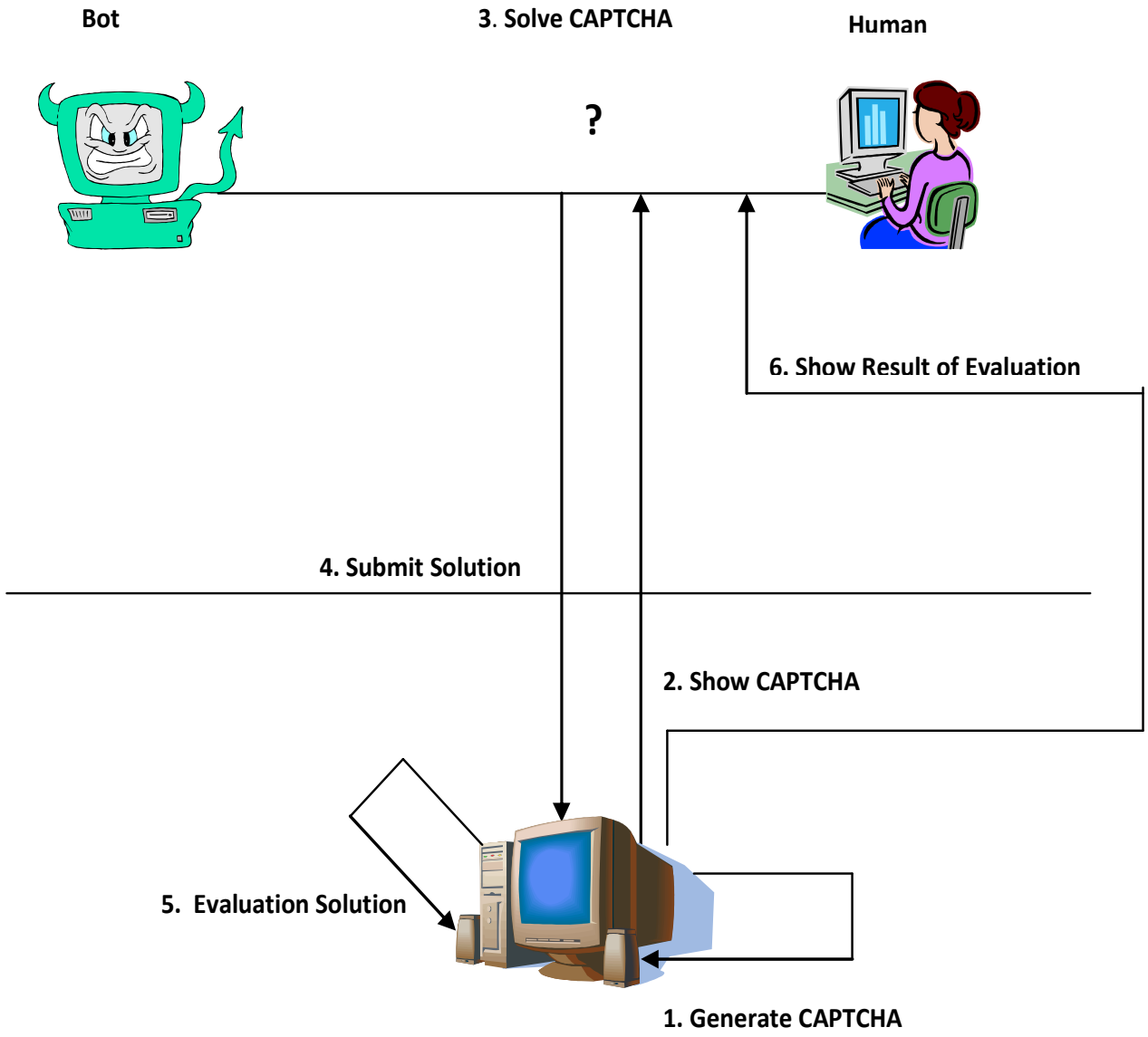

Fig 3: A Typical Captcha Image Generation Implementation Framework

This single response captcha mechanism allows human or robot to retry as many times as possible at the time the user deems fit. This means that the bot has ample of time to send the captcha challenge to spammers who solves the captcha at their own time and the bot captures the solved challenge and then administers it to the site and gets access to whatever information it requires. This is a risk to the websites.

A Prototype of an e-mail service scenario, which is a specific instance of a single challenge system, is depicted in figure 4.

The following is obtainable in an e-mail service scenario depicted in figure 4:

1. The user makes a request such as request to open an email account or fill a form on an email account.
2. The email client requests the email server.
3. The email server asks the CAPTCHA server to generate a CAPTCHA test to verify the user: Human/ AI Assistance Program.
4. The CAPTCHA server returns a test and its answer.
5. The email server responds to email client with the CAPTCHA test.
6. The email client calls the CAPTCHA tester to deliver the challenge.
7. The test is shown to the human/AI assistance program.
8. The CAPTCHA tester collects the user's answer.
9. The CAPTCHA tester forwards result to the email client.
10. The email client submits the solution to the email server.
11. The email server evaluates the submitted solution and passes the result to the email client.
12. The email client reports the result of evaluation to the user – human/ AI Assistance Program – and allows or blocks access to a resource based on the result.

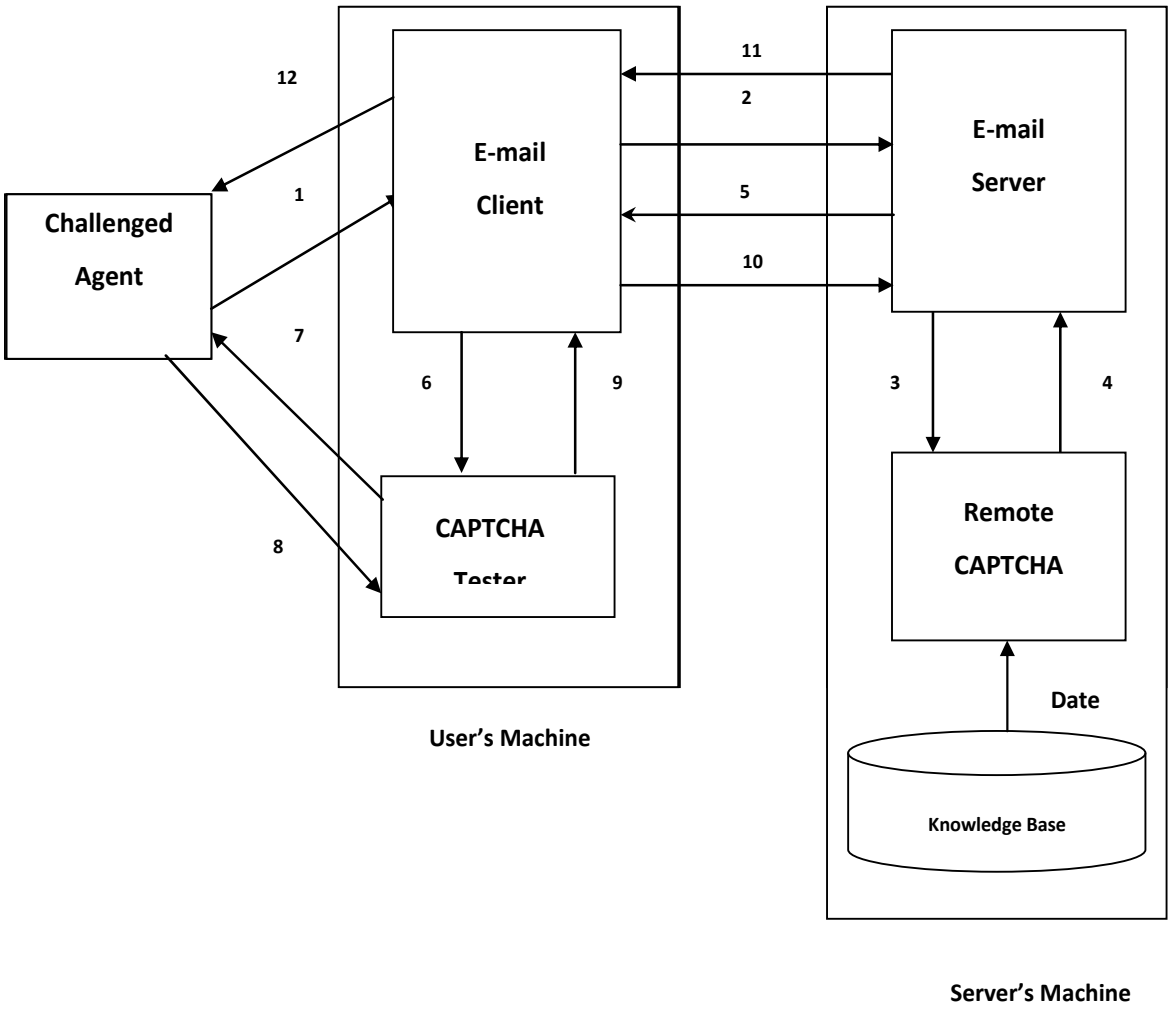

Fig. 4: E-mail service Scenario Framework

A non-response or incorrect response to this test indicates an AI assistance program or bot. The design of the CAPTCHA server is to provide a random CAPTCHA challenge, verify the answer, and return a predicate confirming that the answer is correct. The functionality of the CAPTCHA Tester running on the user's machine is to display a CAPTCHA challenge to the user and gets the corresponding answer.

4. ARCHITECTURE OF THE TM-MCR MECHANISM

The architecture of the proposed TM-MCR mechanism is depicted in Figure 5.

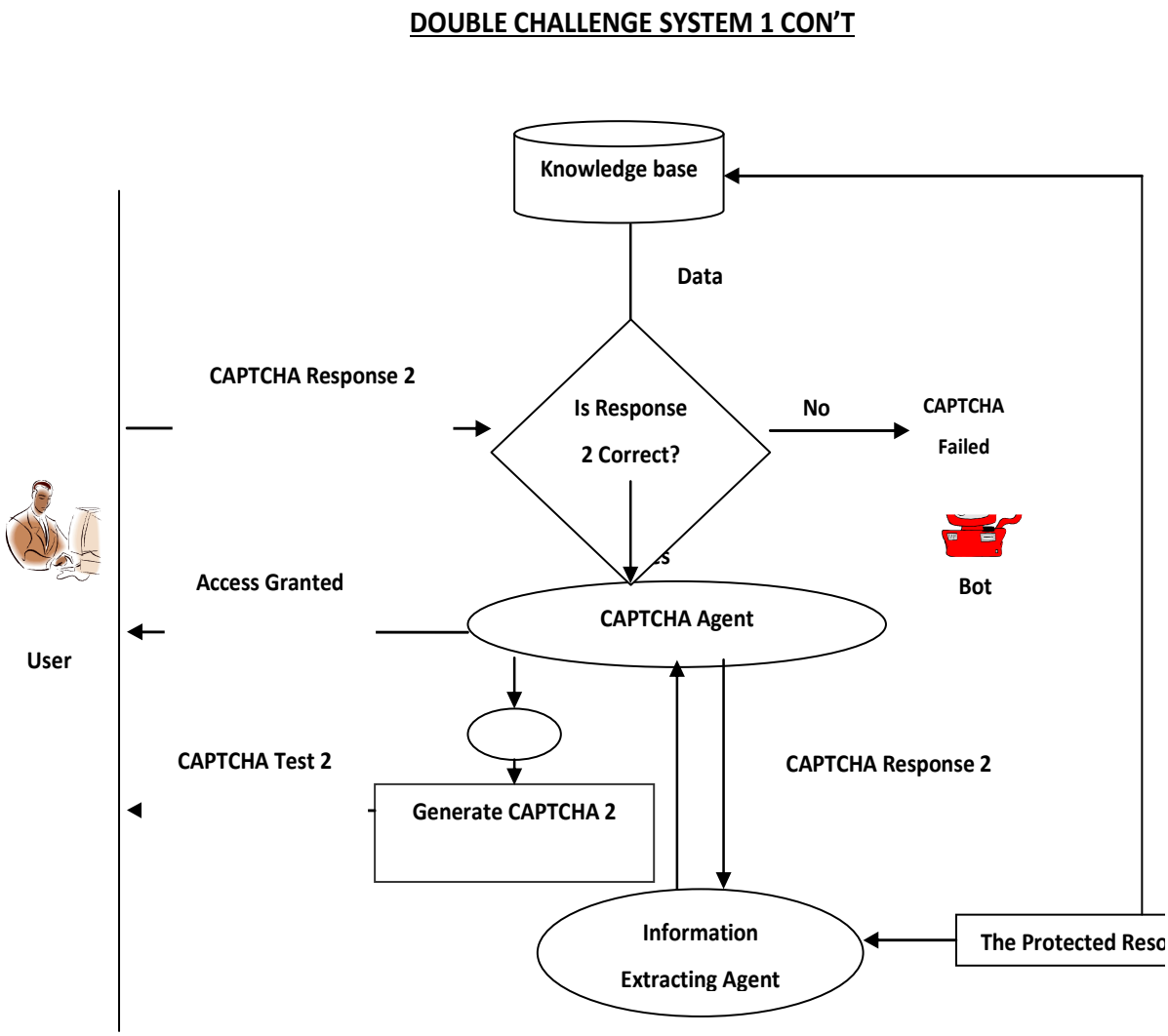

Figure 5: Visual Representation of the Testing Procedure of the Multiple C-R System With Database considered

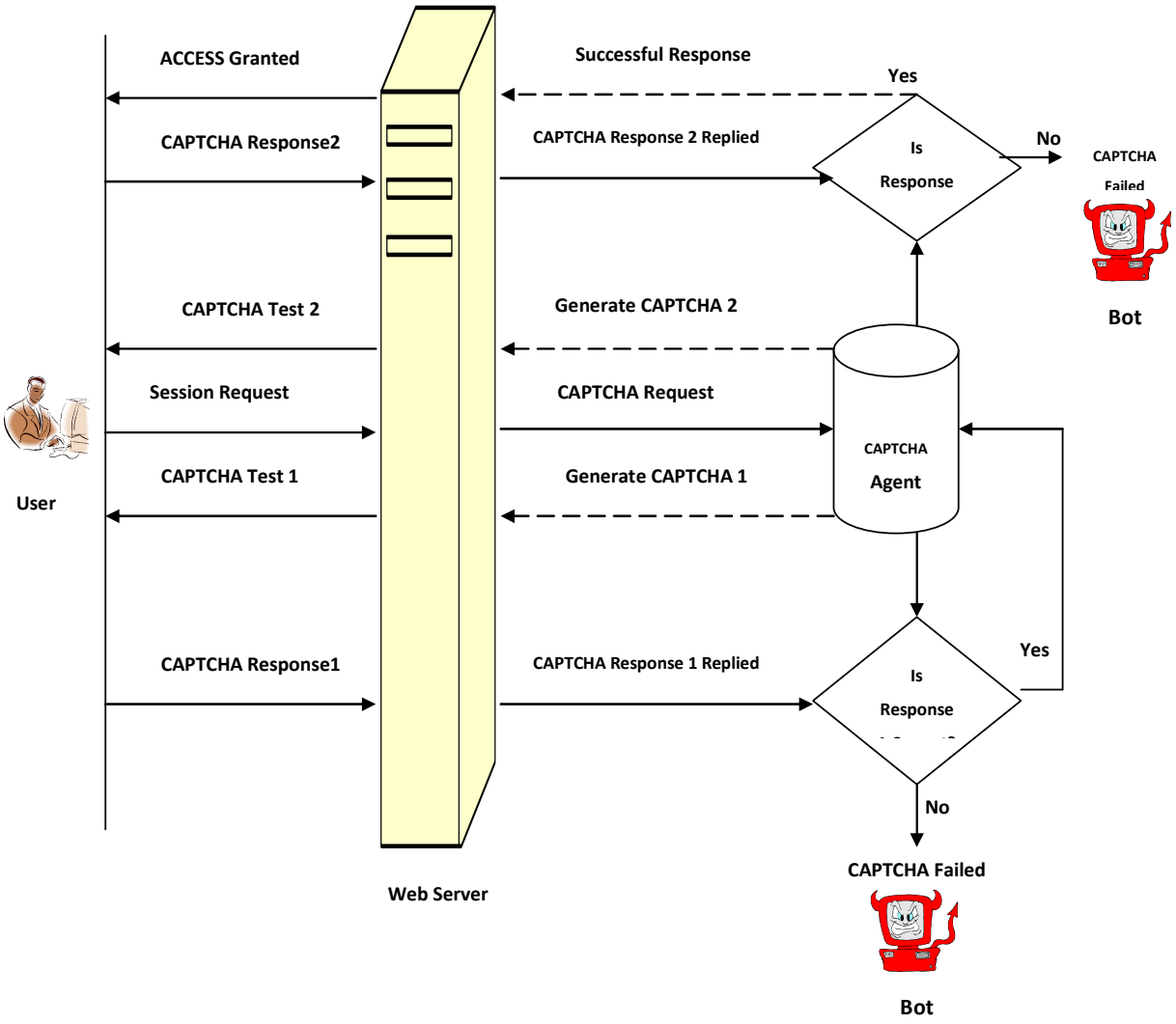

Figure 6: Visual Representation of the Testing Procedure of the Proposed System without Database

5. CONCLUSION

In this work, we have proposed a Time-Out Multiple Challenge Response CAPTCHA that inculcates integrated mathematical mechanism to boost the security of CAPTCHA codes. This is to enable users overcome the limitations of the current CAPTCHA system that consist of only alphanumeric characters. The introduction of special features or characters (such as parentheses, brackets, braces and mathematical characters like Boolean characters) into the code is necessary to strengthen the CAPTCHA's security Mechanism.

6. FUTURE WORK

Future work will implement the proposed mechanism using appropriate software tools.

REFERENCES

[1] Waziri V.O. (2010). Curbing Cyber Crime and the Impact of Cryptography. Proceedings of the Nigerian Computer Society, Vol 21. Pp28-45

[2] Chellapilla K. and Simard P. (2004). Using machine learning to break visual human interaction proofs (HIPs). In Proceedings of the 17th Advances in Neural Information Processing Systems Conference (NIPS '04), MIT Press, Vancouver, Canada.

[3] Athanasopoulos E. & Antonatosn S. (2006). Enchanced CAPTCHAs: Using Animation to Tell Humans and Computers Apart. *LNCS*, 4237, pp. 97-108.

[4] SHA1 Encryption Algorithm. http://www.mailhide.recaptcha.net.

[5] Challenge-response authentication www. Wikipedia.com

[6] Bandaya T. & Shah N.A. (2009). Image Flip CAPTCHA. ISC Int'l Journal of Information Security Germany. Vol. 1.

[7] Mori G. and Malik J. (2003). Recognizing Objects In Adversarial Clutter: Breaking A Visual CAPTCHA. Proceedings of the 2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. I-134–I-144.

[8] Aboufadel E.F., Olsen J. and Windle J. (2005). Breaking the Holiday Inn Priority Club CAPTCHA. The College Mathematics Journal, Vol. 36, No. 2, 2005.

[9] Bentley J. and Mallows C.L. (2006). CAPTCHA Challenge Strings: Problems And Improvements. Proceedings of the Document Recognition and Retrieval, pp. 141–147, San Jose, Calif, USA.

[10] Baird H.S. and Riopka T. (2005). ScatterType: a reading CAPTCHA resistant to segmentation attack. The 12th IS&T/SPIE Conference on Document Recognition and Retrieval (DRR '05), Proceedings of SPIE, pp. 197–207, San Jose, Calif, USA.

[11] Baird H.S., Moll M.A. and Wang S.Y. (2005). A Highly Legible CAPTCHA That Resists Segmentation Attacks. In Proceedings of the 2nd International Workshop on Human Interactive Proofs (HIP '05), vol. 3517, pp. 27–41, Bethlehem, Pa, USA.

[12] Chalmers D.J. (2010). The Singularity: A Philosophical Analysis. Journal of Consciousness Studies, Vol. 17, No. 9-10, pp. 7–65, 2010.

[13] Shulman C.M. (2009). "Arms control and intelligence explosions," in Proceedings of the 7th European Conference on Computing and Philosophy, Barcelona, Spain.

[14] Blum M. (2000). The CAPCTHA Project, Completely Automatic Public Turing Test to Tell Computers and Humans Apart", Dept. of Computer Science, Carnegie- Mellon University. Available online at http://www.captcha.net

[15] CAPTCHAS http://www.captcha.net

[16] Online Polls on the Best Graduate School in Computer Science. Available online at http://www.slashdot.org